design ideas

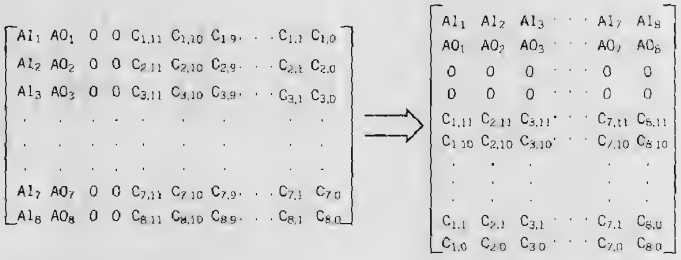
Edited by Bill Travis and Anne Watson Swager

Control 32 DAC channels via a parallel port

Mark A Shill, Burr-Brown Corp, Tucson, AZ

Occasionally, a system needs several digitally programmable voltage-output channels. Such output channels typically provide the control for robot positioning, industrial processes, and even home automation. The circuit in **Figure 1** (pg 94) controls 32 voltage-output channels from the parallel port of a PC. The circuit comprises eight DAC7615 quad voltage-output, serial-data programmable, 12-bit DACs. The controlling PC individually programs each of the 32 DAC channels, and all DAC outputs simultaneously update.

Figure 2



A series of eight quad-output voltage DACs provide 32 channels that you can individually program. All of the DACs simultaneously update.

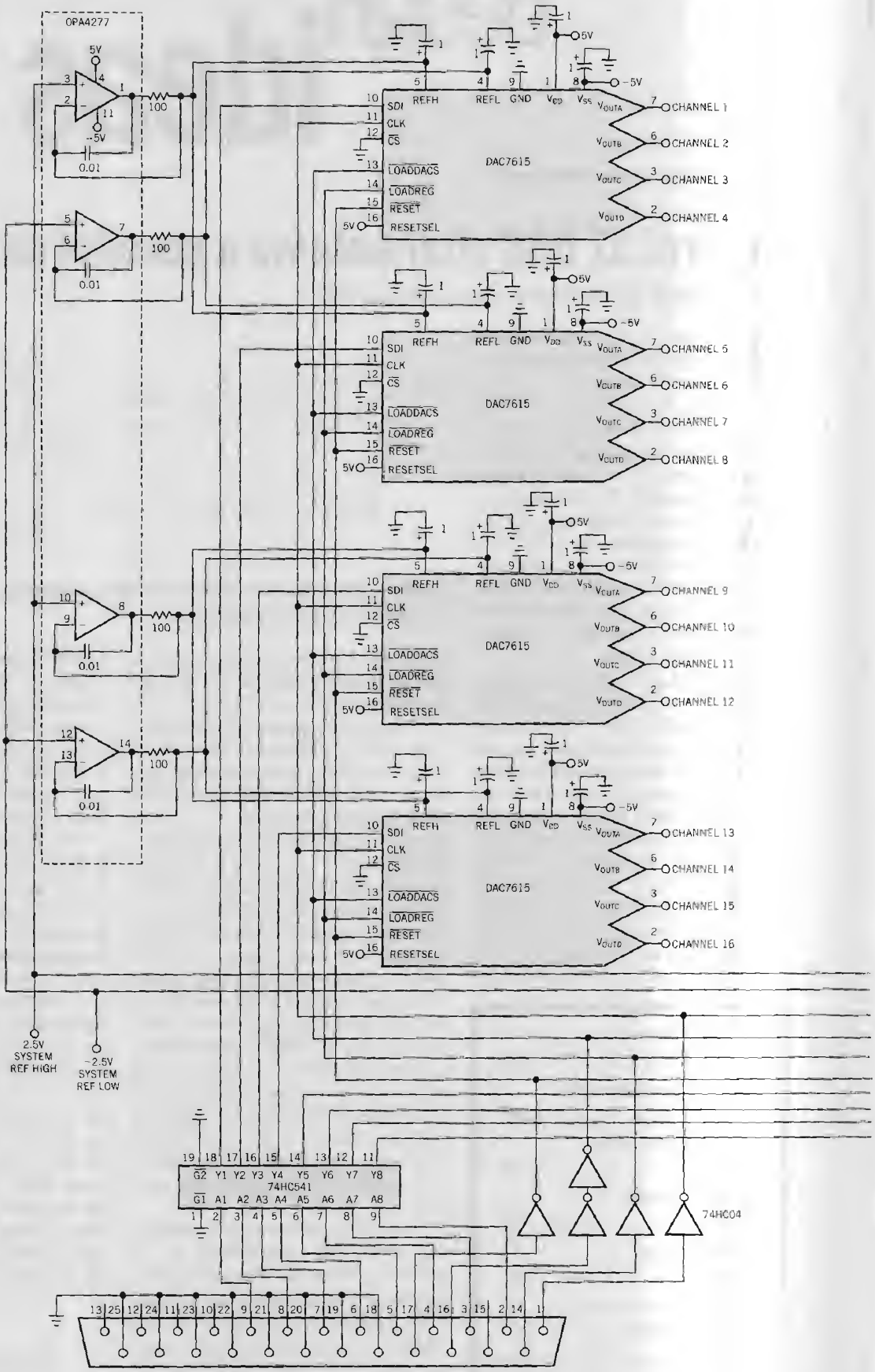
The parallel port's eight data-output lines provide serial data into each of the eight quad DAC7615s. The remaining four control lines of the parallel port provide the serial-data clock, input-register clock, DAC-register clock, and DAC-reset functions. Each DAC7615 has a reference high and low input, which the circuit connects to external reference voltages of 2.5V and −2.5V, respectively. Two OPA4277 quad op amps buffer the ±2.5V DAC reference voltages. Because all of the DACs use the same ±2.5V reference voltages, all DAC outputs track together as a function of these references. The resulting DAC output-voltage range for all 32 channels is −2.5V to +2.5V.

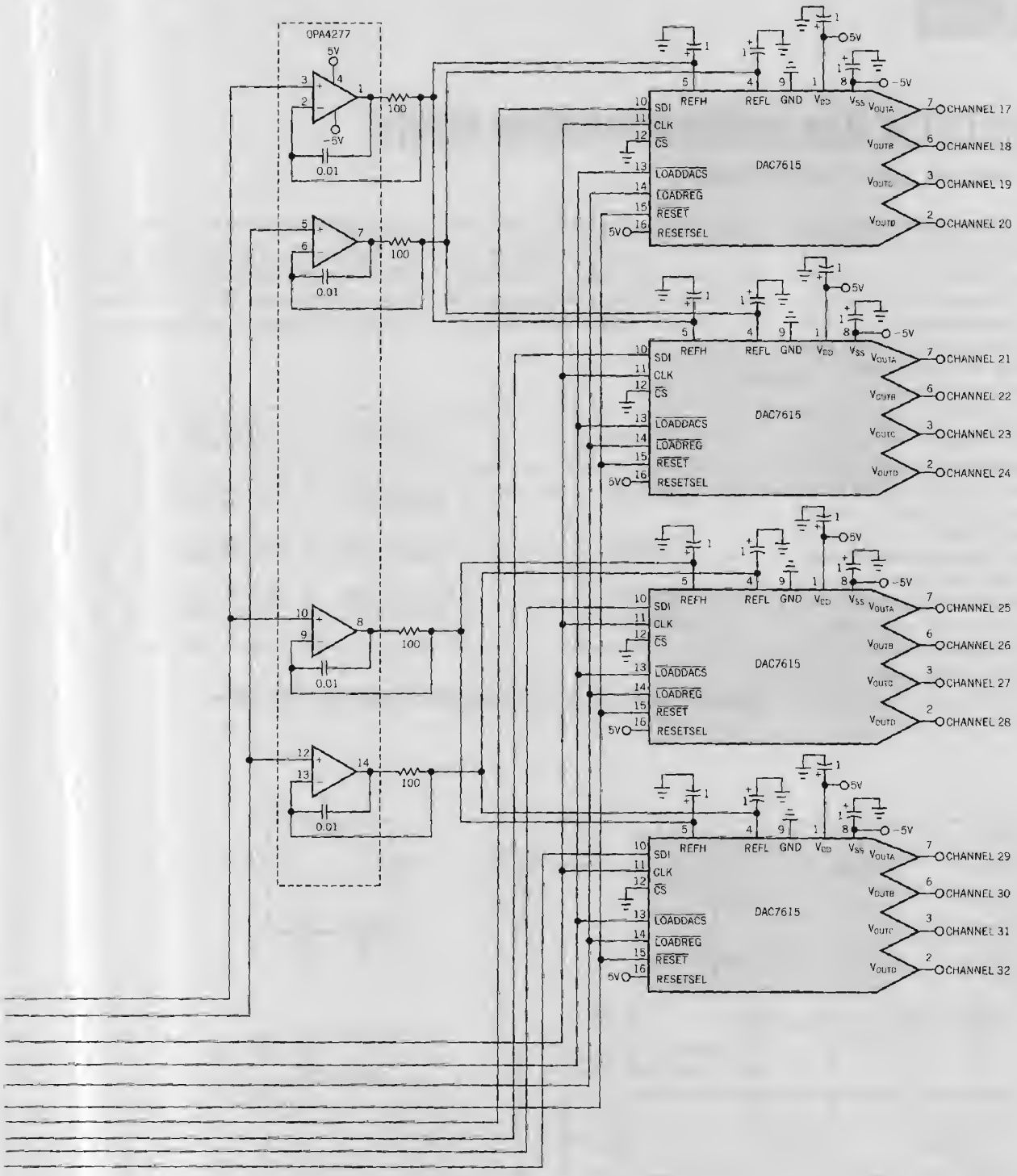
The circuit programs each of the eight DAC7615s by shifting in a serial 16-bit word comprising two address bits, two dummy bits, and the DAC 12-bit data word. The serial data for the V_{OUTA} channel of each DAC7615 shifts in first, followed by the V_{OUTB}, V_{OUTC}, and V_{OUTD} channels. The DAC7615s have a double-buffered data input, so the circuit can load the programmed data for all DAC channels into input registers without changing the previously set DAC output voltage. After each 16-bit word shifts into the corresponding DAC7615, the DAC control line $\overline{\text{LOADREG}}$ momentarily pulses low to latch the shifted data into each DAC's internal input register. Finally, when the circuit has programmed all DAC input registers, the signal $\overline{\text{LOADDACS}}$ pulses low to update the internal DAC registers and change all DAC outputs.

To use the parallel port for simultaneous serial data transmission to all DAC7615s, the software must first manipulate the digital output data to place it in a form that can stream out the parallel port. The controlling software transposes a group of eight 16-bit words, representing the codes to shift into each DAC7615, into a group of 16 8-bit words (**Figure 2**). The resulting vector of 16 8-bit words represents the 16-bit serial data stream, which the circuit simultaneously shifts into the selected one-of-four registers of the DAC7615s. This transposition repeats four times to program all four channels of each DAC7615.

The accompanying program "WriteDAC32," which is written in Borland Turbo Pascal, accepts an array of 32 12-bit codes for programming each of the DAC channels. WriteDAC32 uses an assembly-language procedure to repeatedly left-shift the leading bit of each 12-bit DAC code and then reconstruct 12 8-bit words representing the stream data that the PC outputs on all eight of the parallel port's data lines. To program all 32 DAC channels, 4 × 16 data-clock cycles are necessary. If you daisy-chain the DACs, the number of necessary clock cycles is 4 × 8 × 16. You can download the program from *EDN*'s Web site, www.ednmag.com. Click on "Search Databases/Links Page" and then enter the Software Center to download the file from DI-SIG, #2347. (DI #2347)

Figure 1


OPA4277
5V
-5V
100
0.01
DAC7615
REFH
REFL
GND
VDD
VSS
SDI
CLK
CS
LOADDACS
LOADREG
RESET
RESETSEL
VOUTA
VOUTB
VOUTC
VOUTD
CHANNEL 1
CHANNEL 2
CHANNEL 3
CHANNEL 4
CHANNEL 5
CHANNEL 6
CHANNEL 7
CHANNEL 8
CHANNEL 9
CHANNEL 10
CHANNEL 11
CHANNEL 12
CHANNEL 13
CHANNEL 14
CHANNEL 15
CHANNEL 16
2.5V SYSTEM REF HIGH
-2.5V SYSTEM REF LOW
74HC541
74HC04



NOTE: ALL CAPACITORS ARE IN MICROFARADS.

The controlling software transposes eight 16-bit words into a vector of 16 8-bit words.

Comparator has programmable limits

Michele Frantisek, Brno, Czech Republic

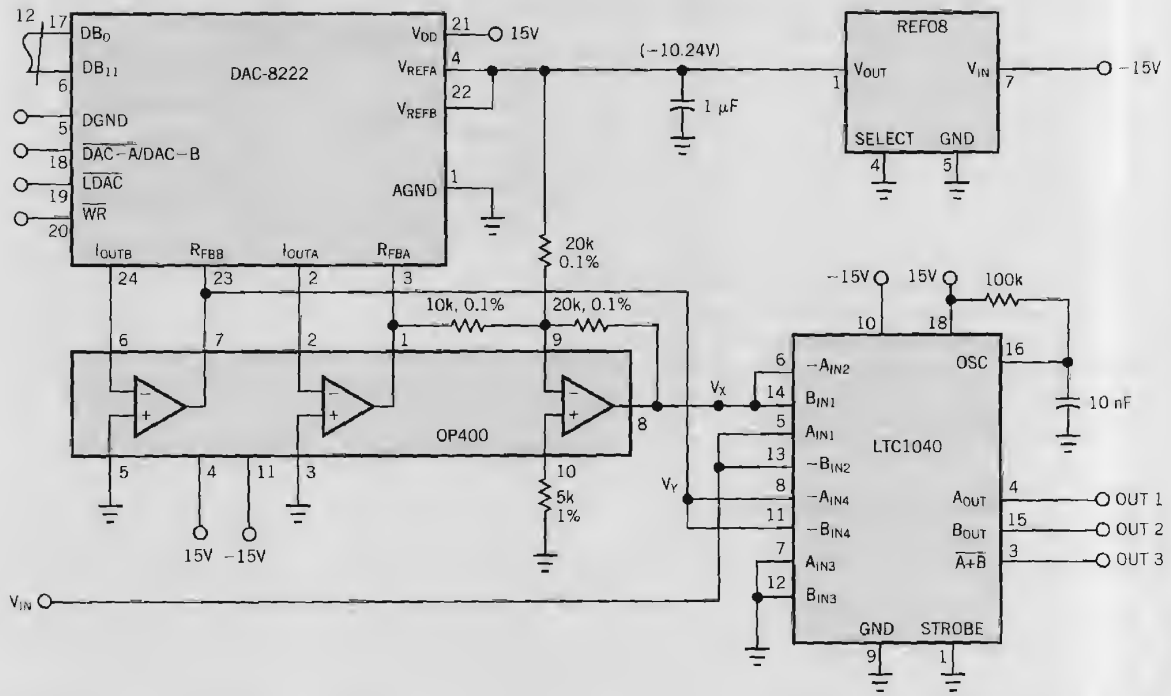
THE CIRCUIT IN **Figure 1** combines a dual buffered D/A converter and dual four-input comparator to configure a comparator circuit with a digitally programmable window center and width. The circuit has three outputs that separately indicate the comparison states: within the window, over the upper limit, or below the lower limit. With the component values shown, you can program the center voltage of the window from −10.24 to +10.235V in 5-mV steps, and the width of the window from 0 to 20.47V, also in 5-mV steps. The programmed values are fully independent of each other and of the input voltage. The dual buffered DAC-8222, together with three op amps from an OP-400, generates the voltages V_X (center voltage for the LTC1040) and V_Y (half the window width) from binary data stored in the DAC's latches. Whereas DAC A of the DAC-8222 operates in a bipolar configuration, DAC B operates in unipolar mode. A μP's address bus generates the DAC's control signals $\overline{\text{DAC-A/DAC B}}$, LDAC, and WR. The reference voltage for the V_{REFA} and V_{REFB} inputs of the DAC-8222 comes from the REF08, configured for a −10.24V output. The LTC1040 contains two sampling-mode comparators

Figure 2

FORM OF WINDOW	V_X	V_Y	LATCHED INPUT DATA IN DAC A	LATCHED INPUT DATA IN DAC B
1V / −1V	0V	1V	1000 0000 0000 DECIMAL 2048	0001 1001 0000 DECIMAL 400
2V / 0V	1V	1V	0111 0011 1000 DECIMAL 1848	0001 1001 0000 DECIMAL 400
2V / −4V	−1V	3V	1000 1100 1000 DECIMAL 2248	0100 1011 0000 DECIMAL 1200

Examples of V_X and V_Y and their digital equivalents are stored in the DAC's latches.

Figure 1



A precision comparator uses a D/A converter to provide precise, noninteractive control of the upper and lower thresholds.

that drive the three outputs of the circuit. Output Out 1 assumes a logic-high state if the algebraic sum of the voltages at A_{IN1}, A_{IN2}, A_{IN3}, and A_{IN4} is positive, as the following equations show: $V_{IN}-V_X>0$; therefore, $V_{IN}>V_X+V_Y$. Out 1 thus assumes a high state if V_{IN} is greater than the upper limit of the window (V_X+V_Y).

Similarly, Out 2 assumes a high state if the sum of voltages B_{IN1}, B_{IN2}, B_{IN3}, and B_{IN4} is positive: $V_X-V_{IN}-V_Y>0$; therefore, $V_{IN}<V_X-V_Y$. Thus, Out 2 goes high if the value of V_{IN} is lower than the bottom limit of the window (V_X-V_Y). Finally, Out 3 assumes a high state if both Output 1 and Output 2 are low: $V_X-V_{YY}<V_{IN}<V_X+V_Y$. Thus, Output 3 goes high if the value of V_{IN} is greater than the lower limit and lower than the upper limit of the window. The RC combination at Pin 16 of the LTC1040 determines the sampling rate; in this case, approximately 1000 samples/sec. **Figure 2** gives some examples of V_X and V_Y and their digital equivalents stored in the DAC's latches. The circuit needs no calibration and produces maximum errors of ±10 mV over the full range. (DI #2377)

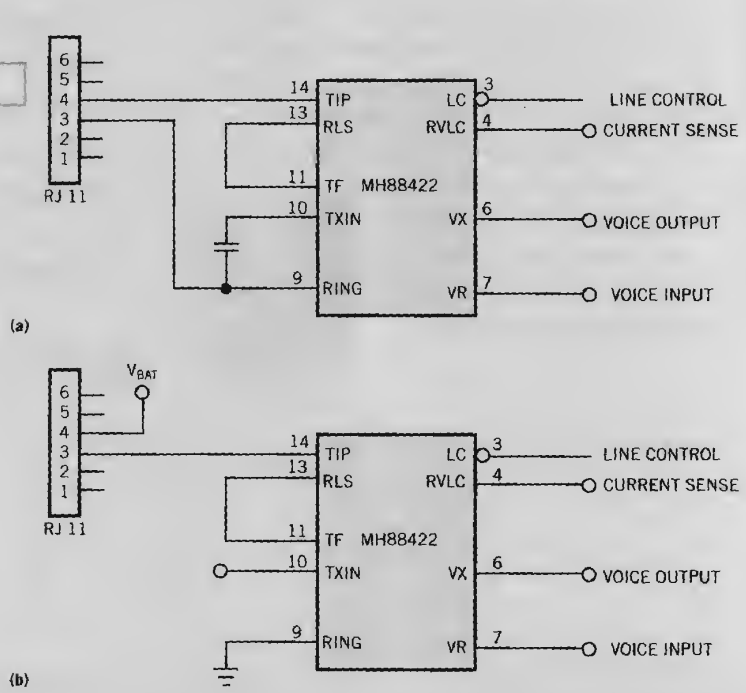
To Vote For This Design, Circle No. 347

DAA circuit emulates central-office operation

Jerzy Chrzaszcz, Warsaw University of Technology, Poland

The Mitel MH88422 data-access arrangement (DAA), a thick-film hybrid module, contains a complete interface between duplex voice- or data-transmission equipment and an analog telephone line. It provides transformerless, optoisolated two-to-four-wire conversion with transhybrid loss cancellation and operates from a 5V supply. The DAA also consumes low on-hook power. **Figure 1a** gives Mitel's typical application circuit. The Tip and Ring lines connect to a central-office or private-branch-exchange line, and the interface mimics the operation of a telephone set. The modification in **Figure 1b** changes the function of the interface such that an ordinary single-line telephone can connect directly to such systems as a voice/touch-tone peripheral device.

Figure 1



A modification (b) of a DAA's recommended interface (a) provides central-office-like operation for single-line telephones.

In this configuration, line-control (LC) input is always active. When the telephone goes off-hook, the path from battery to ground closes. Line current flowing in the sense resistor in the MH88422 activates ring voltage/line current (RVLC), thereby signaling the off-hook state to the system controller. Analog functions are as in the original configuration; that is, you can transmit and receive signals. The modified system provides no ring signal and thus can serve only incoming calls; nevertheless, the interface operates much like a central-office arrangement. (DI #2376).

To Vote For This Design, Circle No. 348

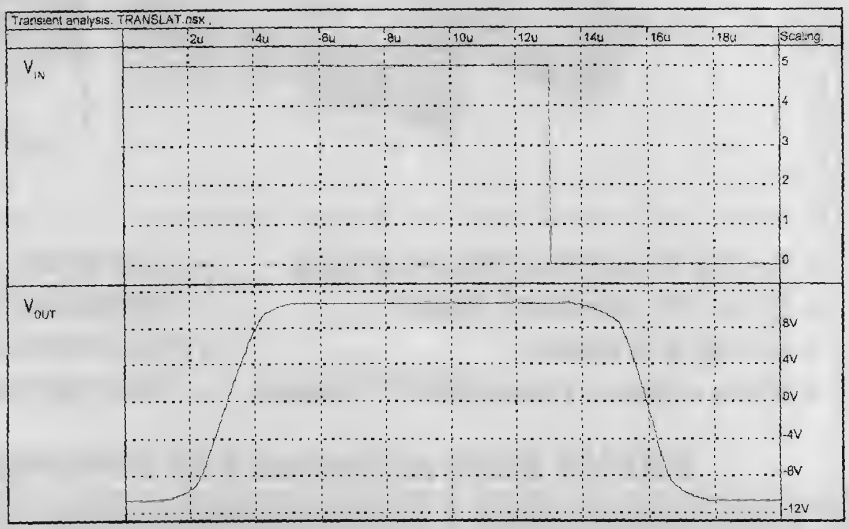
Speedy logic translator uses little power

Davis Magliocco, CDPI, Scientrier, France

IN HANDHELD EQUIPMENT, component count and power consumption are critical considerations. This Design Idea uses only a few transistors to configure a high-speed output stage of an RS-232C link while draining few precious milliamperes from the batteries. In general, the data to transmit comes from an IC (for example, an ADC or a μC) connected to 5V and ground. The RS-232C protocol requires ±12V transmission levels. You can derive the voltages directly from the RS-232C line of a computer with a simple diode circuit. You need a charge pump only in the case of a direct connection to a printer because the 12 and −12V may not be available simultaneously.

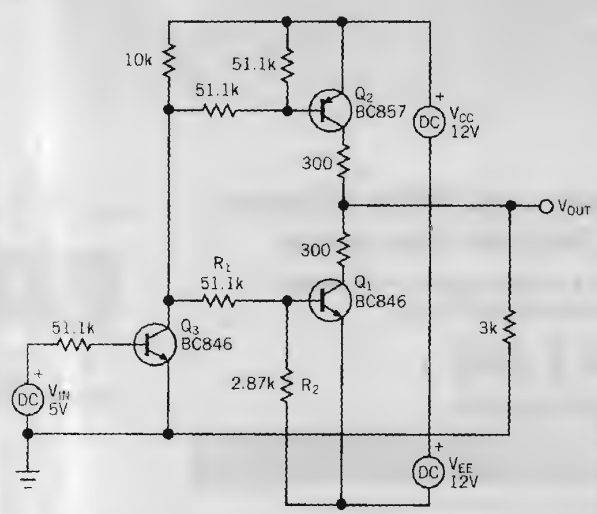
The translation from ±12V to 0/5V of the signals coming from the RS-232C link is easy to implement with a standard Schmitt trigger, such as a CD40106 or CD4584, with protection resistors in series with the inputs. The circuit in **Figure 1** translates in the other direction. The circuit has some shortcomings: It wastes power through R_1 and R_2 turning off Q_1 while Q_2 is on. Also, the time constants

Figure 3



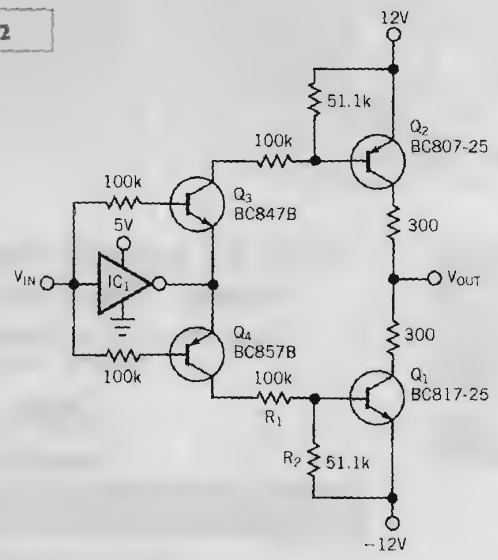
Input-output delay and rise and fall times are all approximately 2 μsec in Figure 2's circuit.

Figure 1



This circuit provides TTL-to-RS-232C translation but suffers from high power consumption and dissimilar rise and fall times.

Figure 2



A simple logic gate cuts power drain and provides high-speed operation with symmetrical rise and fall times.

differ in turning Q_1 and Q_2 on; the output rise time can be 50% longer than the fall time. To obtain high speed and avoid transmission errors stemming from wrong bit length, you must lower all the resistor values, thereby raising power consumption.

The circuit in **Figure 2** draws current only when turning Q_1 or Q_2 on. When the output of IC_1 is low, Q_3 is on, and the reverse-biased Q_4 is off; no current flows in R_1 and R_2. When the output of IC_1 goes high, Q_3 turns off while Q_4 turns on. Because the two branches are symmetrical, the time constants are similar, and the output exhibits similar rise and fall times (**Figure 3**). With one-tenth the power consumption of the circuit in **Figure 1**, the circuit achieves the same transmission speed without risk of bit-length error from asymmetrical rise and fall times. (DI #2378)

Parallel port replaces embedded μC

Eli Kohav and Leonid Grossman, ECI Telecom, Petah-Tikva, Israel

Many applications use an embedded processor, which has certain needs: software, RAM, ROM, board space, and others. Frequently, another host computer, usually a PC, controls the application. Using a single CPLD, you can dispense with the embedded processor and let the PC directly control your system via the PC's parallel port (**Figure 1**). The CPLD mimics the address, data, and control buses of a standard μC, such as an 8052, so you can use standard μC interfaces and peripherals in your system. The advantages you glean from this arrangement are:

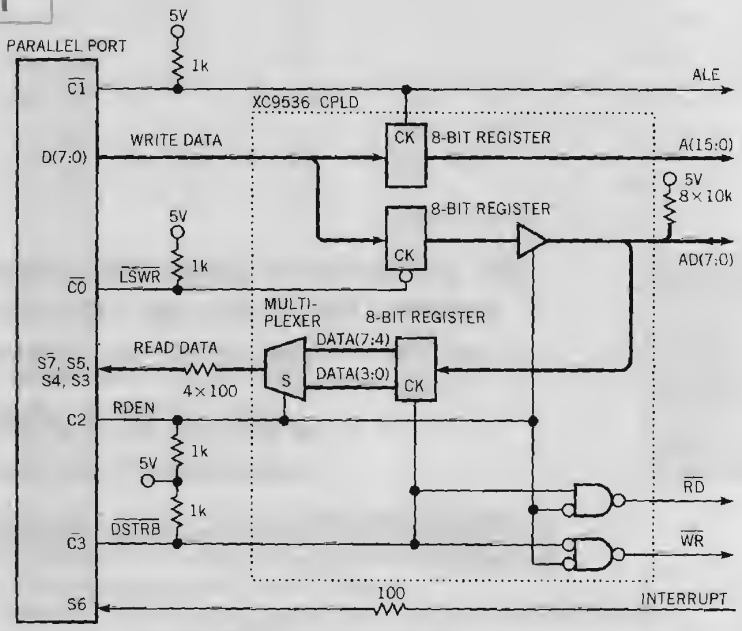
- You need no special development software for your embedded processor. You can use standard PC software. Thus, you need not develop software for two systems.
- You can dispense with the embedded μC; its ancillary RAM, ROM, and crystal oscillator; and other components.
- You free up space on your system's pc board.
- You can benefit from the PC's computing power and its huge base of ready-made software.

The circuit in **Figure 1** uses a standard parallel-port interface, which has 12 digital outputs and five digital inputs, which you access via three successive 8-bit ports in the PC's I/O space: Data Port—eight output pins: D(7:0); the Control Port—four outputs (three inverted): $\overline{C3}$, C2, $\overline{C1}$, and $\overline{C0}$; and the Status Port—five inputs (one inverted): $\overline{S7}$, S6, S5, S4, and S3.

The design fits in a small CPLD: Xilinx's XC9536. The CPLD has three internal 8-bit registers. One latches the high-address bus—A(15:8). The other two latch the multiplexed low-address/Write data, and the Read data—AD(7:0). The PC writes address and data via the Data Port and reads data via the Status Port. Status Pin S6 provides an interrupt. The Control Port generates the control signals of the emulated processor, as well as the internal control signals of the CPLD. To implement a write cycle, the PC issues three successive bytes on the Data Port: the low-order address byte, the high-order address byte, and the data byte. The Control Port generates the needed control signals (**Figure 2a**). The system implements the read cycle by issuing the address in the same order and then reads the data byte in two cycles: low-order data nybble and high-order data nybble (**Figure 2b**).

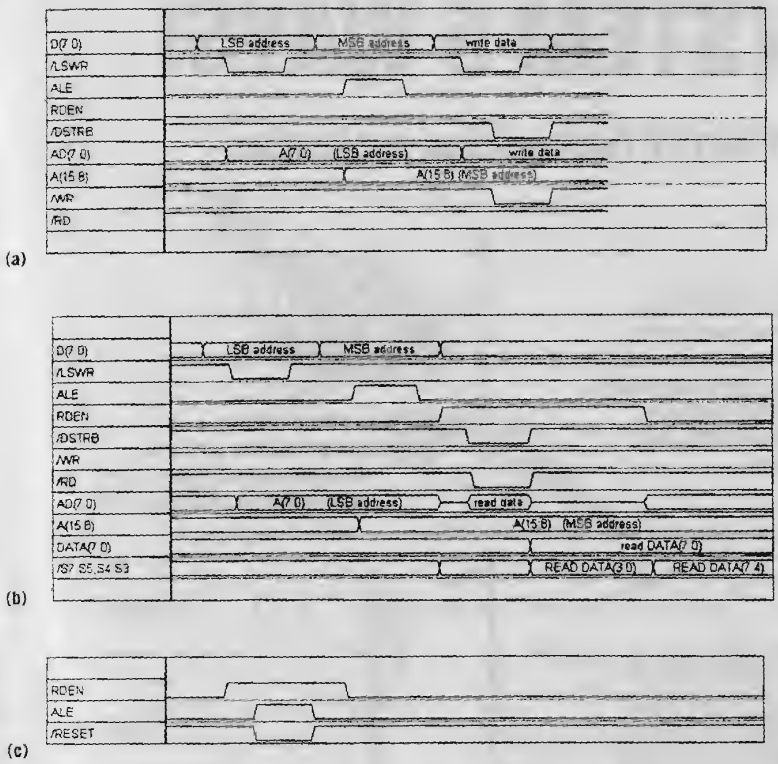
Note that, although the procedure is relatively slow, it's fast enough for most applications, because total I/O access is only a fraction of the application-soft-

Figure 1



A CPLD and a PC's parallel port save component count and board space by replacing an embedded processor.

Figure 2



A PC takes over a µC's job in controlling the write (a) and read (b) cycles in an application; an optional Reset generator (c) requires a somewhat larger CPLD.

Figure 3 (OPTIONAL)

You can exploit otherwise unused control signals in Figure 1's configuration to generate a Reset signal.

ware time budget. The serially connected 100Ω resistors degrade the slew rate of the signals routed to the PC to prevent transmission-line effects on the parallel port's cable. The 1-kΩ pullup resistors connect to the open-collector signals on the Control Port. The 10-kΩ pullup resistors eliminate floating conditions on the AD(7:0) bus. This application needs no Reset signal. If your design needs one, you can generate it by using a memory-mapped port or an unused combination of the control signals (**Figures 2c** and **3**). In this case, you need a larger CPLD. (DI #2374)

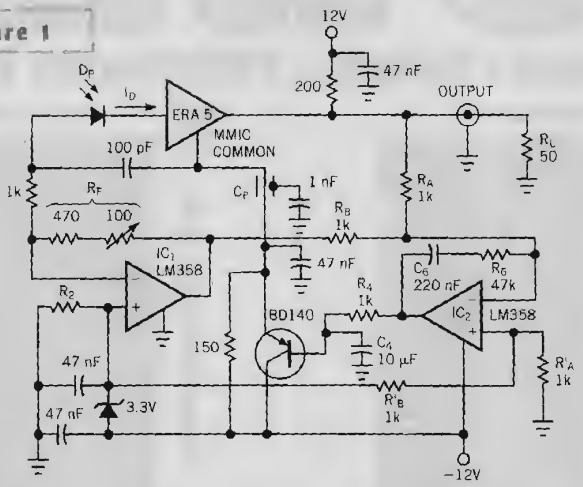
Transimpedance amp covers dc to gigahertz range

Lukasz Sliwczyński and Przemysaw Krehlik,
University of Mining and Metallurgy, Kraków, Poland

To convert the weak, broadband signal from a fiber-optic transmission channel into electrical form, you can use a high-impedance receiver or a transimpedance amplifier. Either method provides the desired gain and bandwidth but removes any dc and low-frequency components of the signal, because both methods require ac coupling. In a situation in which you need to also amplify the dc component, neither method is satisfactory. The feedforward compensation scheme in **Figure 1** solves the problem. The circuit exploits an ERA 5 monolithic-microwave IC (MMIC) from Mini-Circuits (Brooklyn, NY, www.minicircuits.com) as the RF amplifier. It's easier to use such an off-the-shelf part than to configure your own gigahertz-region amplifier.

You can consider the approach in **Figure 1** as a sort of transimpedance amplifier with addition of a feedforward compensation path. The circuit processes signals from 0 Hz to the high-frequency cutoff of the MMIC (approximately 4 GHz for the ERA 5). Current from the reverse-biased photodiode, D_p, an InGaAs C 30617BQC PIN diode from EG&G Canada (www.egginc.com), enters the MMIC at 50Ω input impedance, virtually a short circuit for the photodiode. The MMIC converts the current to a voltage, which appears on the load resistor, R_L. The MMIC exhibits an output-offset voltage of approximately 5V, referred to the MMIC's common pin. IC_2 closes a feedback path around the MMIC, thus removing any offset voltage between the MMIC and circuit ground. You could say that IC_2 substitutes for an output capacitor.

Figure 1



Feedforward compensation allows you to preserve the dc and low-frequency components of fiber-optic transmissions.

Exchanging the feedback circuit for an output capacitor provides a summing point, convenient for introducing the compensating signal. This signal comprises the current flowing into the second photodiode terminal (normally used only for polarization) and the transimpedance amplifier made up of amplifier IC_1 and resistor R_F. The signal routes next through R_B to the summing point at the inverting input of amplifier IC_2. Because the voltage at the summing point must be constant, any dc current from the photodiode must influence the MMIC's output voltage, to compensate the voltage at the output of amplifier IC_1. R_C and C_C cancel the pole resulting from the decoupling filter, R_1-C_4, thus guaranteeing stability.

Assuming that R_A equals R_B, R_A' equals R_B', and the condition $RF=Z_{TO}/2$ is fulfilled, the transimpedance gain of the entire circuit is $k=-(S_p Z_{TO})/2$, where Z_{TO} is the transimpedance of the MMIC and S_p is the photodiode responsivity (typically 0.85A/W at 1310 nm). You can obtain Z_{TO} from the scattering parameters of the MMIC: $Z_{TO}\simeq 2s_{21}Z_O$, where Z_O is the termination resistance for the scattering matrix, usually 50Ω. Z_{TO} for the ERA 5 MMIC is approximately 1 kΩ. If you don't fulfil the foregoing conditions, the frequency characteristics of the circuit will not be flat in the low-frequency region.

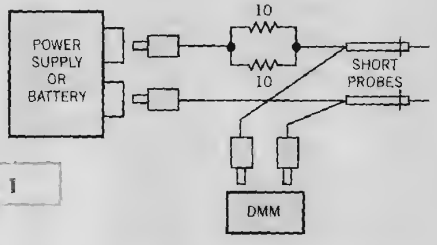
Because you don't know the exact value of Z_{TO}, you should trim R_F for the MMIC you use. You can perform the trim using either a low-frequency spectrum analyzer or a pulse generator with an oscilloscope. With perfect compensation, the frequency response of the amplifier should be flat and should display no overshoot or undershoot. To get the best results, you must take care in designing the pc-board layout because of the circuit's high bandwidth. Especially, decouple the MMIC's common point using components as small as possible to reduce parasitic inductance. You should accurately match resistors R_A-R_A' and R_B-R_B' to minimize offset voltages. (DI #2386).

To Vote For This Design,
Circle No. 351

Simple scheme detects shorts

Luis Miguel Brugarolas, SIRE, Madrid, Spain

Figure 1



A do-it-yourself short-circuit detector makes it easy to find solder bridges in pc board using surface-mount components.

When you manually assemble complex boards, it's common to short-circuit adjacent component or IC pins. Determining which section of the circuit you have shorted is not too difficult, but finding the precise point where the short exists can be a formidable task, because the short may be under a surface-mount component. The circuit in **Figure 1** eases the diagnosis. It uses off-the-shelf components, and you can build it in a few minutes. The circuit uses a DMM set at its maximum voltage-sensitivity scale (typically, 200 mV full-scale with 0.1-mV resolution). The DMM measures the voltage drop in the divider comprising the 5Ω resistor and the cable and short-circuit resistance.

For the power source, you can use a laboratory supply or a battery cell. The low power-source voltage guarantees that no circuit damage can occur, even if you probe the wrong circuit points. For a 1V source, the circuit's transfer function is 0.5 mV/mΩ—enough sensitivity for any practical situation. The accuracy of the DMM is not important, but resolution is. The scheme is simple to use: With the circuit under test unpowered, connect the probes in any area of suspected shorted nets. Move one probe in a direction to minimize the voltage reading. Then, move the other probe to find the point that produces an absolute minimum reading. The short circuit is most likely between the points the two probes touch. (DI #2385)

To Vote For This Design,
Circle No. 352